Updates for Mastering Regular Expressions, 3rd edition

page xvii

Printed: It was published in 2002.
Fixed: The second edition was published in 2002.
Clarifies some poor wording.

page 13, last line of 3rd paragraph

Printed: understand that (first|1st) and (fir|1)st effectively...
Fixed: understand that (First|1st) and (Fir|1)st effectively...
It's not wrong how it was presented, but to match better with the prior discussion, it makes sense to capitalize the 'f'.

page 49, first list item

Printed: Perl's appear to be of the same general type as *egrep*'s, but has a....
Fixed: Perl's appears to be of the same general type as *egrep*'s, but Perl's has a....
page 63, table, 3rd row

Printed: ..., but this not of much practical...
Fixed: ..., but this is not of much practical...
page 69, last line

Printed: ... or perhaps [^ \t\r]*$ to allow for
Fixed: ... or perhaps [^ \t]*$ to allow for
See the next errata item for the description.

page 70, first line

Printed: spaces, tabs, and the carriage return that some systems have before the line-ending newline.
Fixed: spaces and tabs.
Including \r creates problems on the very systems mentioned (removing the \r required at the end of line).

page 70, second paragraph

Printed: [^ \t\r]*$
Fixed: [^ \t]*$ to allow for
See prior item.

page 70, mid page

Printed: $text =~ s/[^ \t\r]*$/<p>/mg;
Fixed: $text =~ s/[^ \t]*$/<p>/mg;
See prior item.

page 74, last line

Printed: maybe some tech stuff at http://www.slashdot.com!
Fixed: maybe some tech stuff at http://www.slashdot.com!
The "!" should be included in the underlined text.

page 94, last line of 2nd paragraph

Printed: style is use by most...

Fixed: style is used by most...
page 96, 2nd code snippet

Printed: if (! line.matches("\\s*",))
Fixed: if (! line.matches("\\s*"))
page 199, mid page

Printed: s/\s*(.*?)\s*$/$1/s
Fixed: s/^\s*(.*?)\s*$/$1/s
page 210, 2nd-to-last paragraph

Printed: ...better to use [1235-9] instead...
Fixed: ...better to use [01235-9] instead...
or ...better to use [0-35-9] instead...
Forgot that zero is a digit. Doh!

page 218, middle of third paragraph

Printed: ..., would I do add this?
Fixed: ..., would I add this?
page 261, third paragraph

Printed: ... the distribution of the colon from (?:this|that):) to ...
Fixed: ... the distribution of the colon from (?:this|that): to ...
page 298, bottom of second paragraph

Printed: ... the whole reason we've been looking localization: ...
Fixed: ... the whole reason we've been looking at localization: ...
page 370, last paragraph

Printed: (it renders LF and all the other non-newline...
Fixed: (it renders CR and all the other non-newline...

page 378, first code snippet

Printed: String regex = "(?x) ^(https?):// ([^/:]+) (?:(\\d+))?";
Fixed: String regex = "(?x) ^(https?):// ([^/:]+) (?: :(\\d+))?";
page 447, last paragraph

Printed: ..., '+' is treated as '*?' and vice-versa
Fixed: ..., '+' is treated as '+?' and vice-versa

page 457, last line

Printed: they retained, ...
Fixed: they are retained, ...
page 459, last line

Printed: the variables are note interpolated at the wrong time.
Fixed: the variables are not interpolated at the wrong time.
page 467, first code snippet

Printed: $parts = preg_split('/\r? \n \r? \n/x', $response, 2);
Fixed: $parts = preg_split('/\r? \n \r? \n/xS', $response, 2);

The parenthetical note just after this snippet refers to the **S** modifier, so it should have been there.

page 473, comment on the if line in the middle of the page

Printed: /* '/' followed by '\' or EOS */
Fixed: /* '\' followed by '/' or EOS */

page 481, mid-page regex

Printed: ^((?:<(\w++)[^>]*+(?<!/)>(?1)</\2>|[^<>]++|<\w[^>]*+/>)*+)$</tt>
Fixed: ^((?:<(\w++)[^>]*+(?<!/)>(?1)</\2>|[^<>]++|<\w[^>]*/>)*+)$</tt>

This is a pretty major error — the use of a possessive greedy quantifier on '[^>]' means that the '/' that follows can never match (since if there were a slash for the '/' to match, the '[^>]' would have consumed it). In an attempt to make the expression a bit more efficient, I made it incorrect, and that's very bad. If PHP had lazy possessive quantifiers, I could use them here, but it doesn't, so the thing to do is to drop the minor efficiency gains from the possessive quantifiers, and use normal greedy quantifiers. That way, it will at least be correct.

page 482, middle paragraph

Printed: The third alternative, <\w[^>]*+/>, matches...
Fixed: The third alternative, <\w[^>]*/>, matches...

page 482, middle paragraph

Printed: As before, the use of a possessive quantifier here may be overkill but it certainly doesn't hurt.
Fixed: *(that sentence removed)*

page 484, top regex

Printed: | <\w[^>]*+/> # *self-closing tag*
Fixed: | <\w[^>]*/>, # *self-closing tag*

page 484, bottom regex

Printed: | <\w[^>]*+/> # *self-closing tag*
Fixed: | <\w[^>]*/>, # *self-closing tag*
